(12) UK Patent Application (19) GB (11) 2 353 918 (13) A

(43) Date of A Publication 07.03.2001

(21) Application No 9920703.7

(22) Date of Filing 03.09.1999

(51) INT CL[7]
H04Q 7/32

(52) UK CL (Edition S )
H4L LDGX

(56) Documents Cited
EP 0930793 A EP 0767426 A WO 96/24231 A

(58) Field of Search
UK CL (Edition R ) G4A AFL , H4L LDGX LED
INT CL[7] G06F 9/445
On-line: WPI, EPODOC, JAPIO

(71) Applicant(s)
Telefonaktiebolaget L M Ericsson
(Incorporated in Sweden)
S-126 25 Stockholm, Sweden

(72) Inventor(s)
Pasi Matti Kalevi Ahonen

(74) Agent and/or Address for Service
Marks & Clerk
4220 Nash Court, Oxford Business Park South,
OXFORD, OX4 2RU, United Kingdom

(54) Abstract Title
**Access rights in a mobile communications system**

(57) A method of testing an executable software component which may be downloaded over a mobile telecommunications network 1 to a mobile host 3 for execution in the mobile host 3. The method comprises transferring a copy of the executable software component from a memory 6,7 in which it resides to an emulator 3 or 8, the emulator 3,8 being able to at least approximately emulate the operation of a mobile host 3 which wishes to run the software component. The software component is executed in the emulator 3,8 and the security exceptions resulting from execution of the software component are identified.

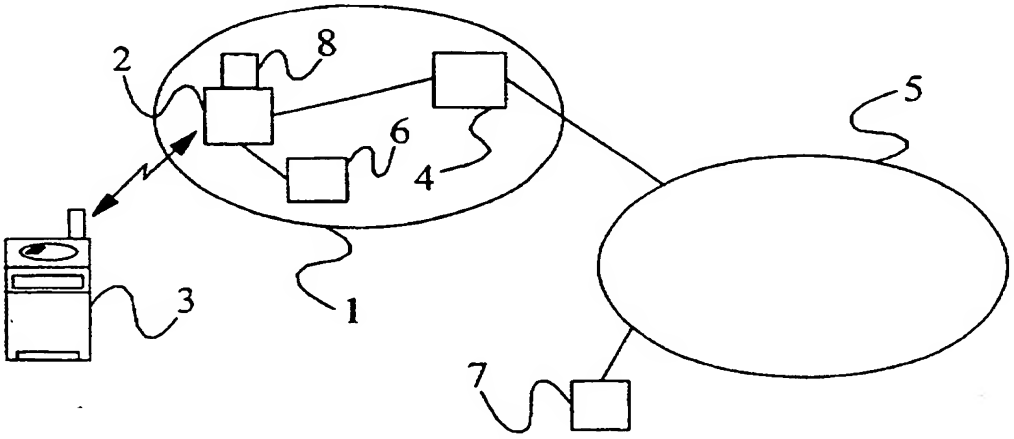


Figure 1

GB 2 353 918 A

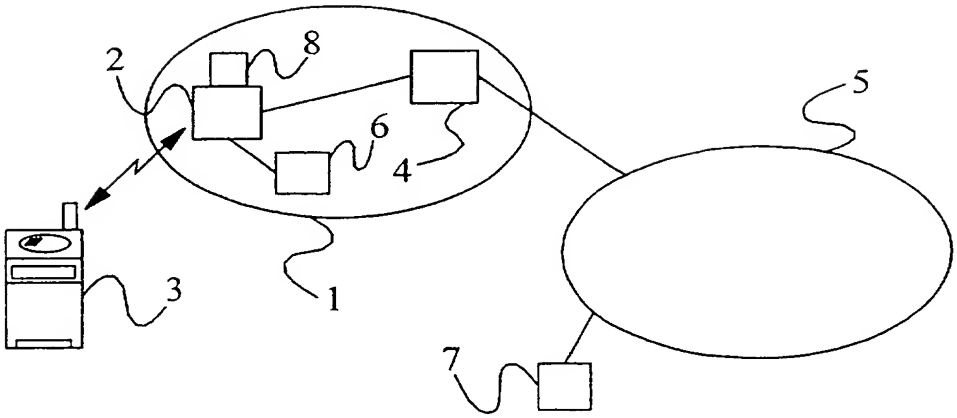


Figure 1

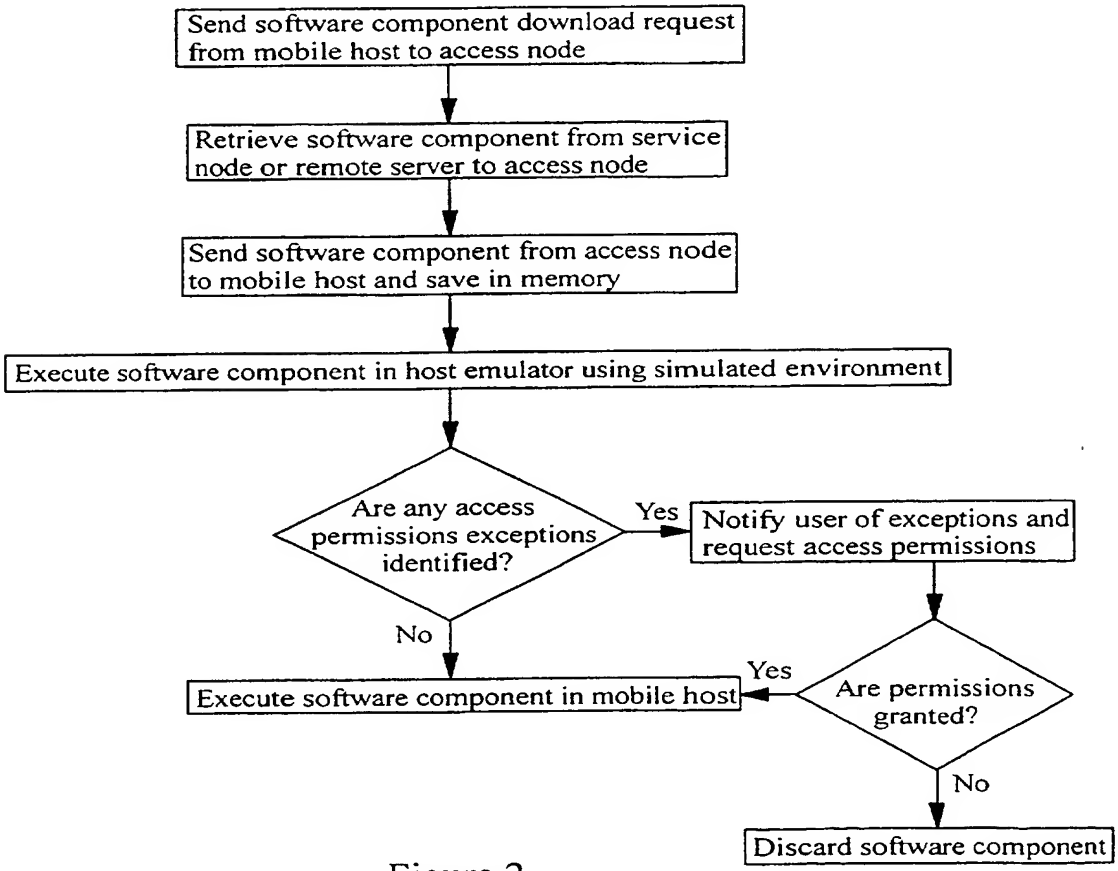


Figure 2

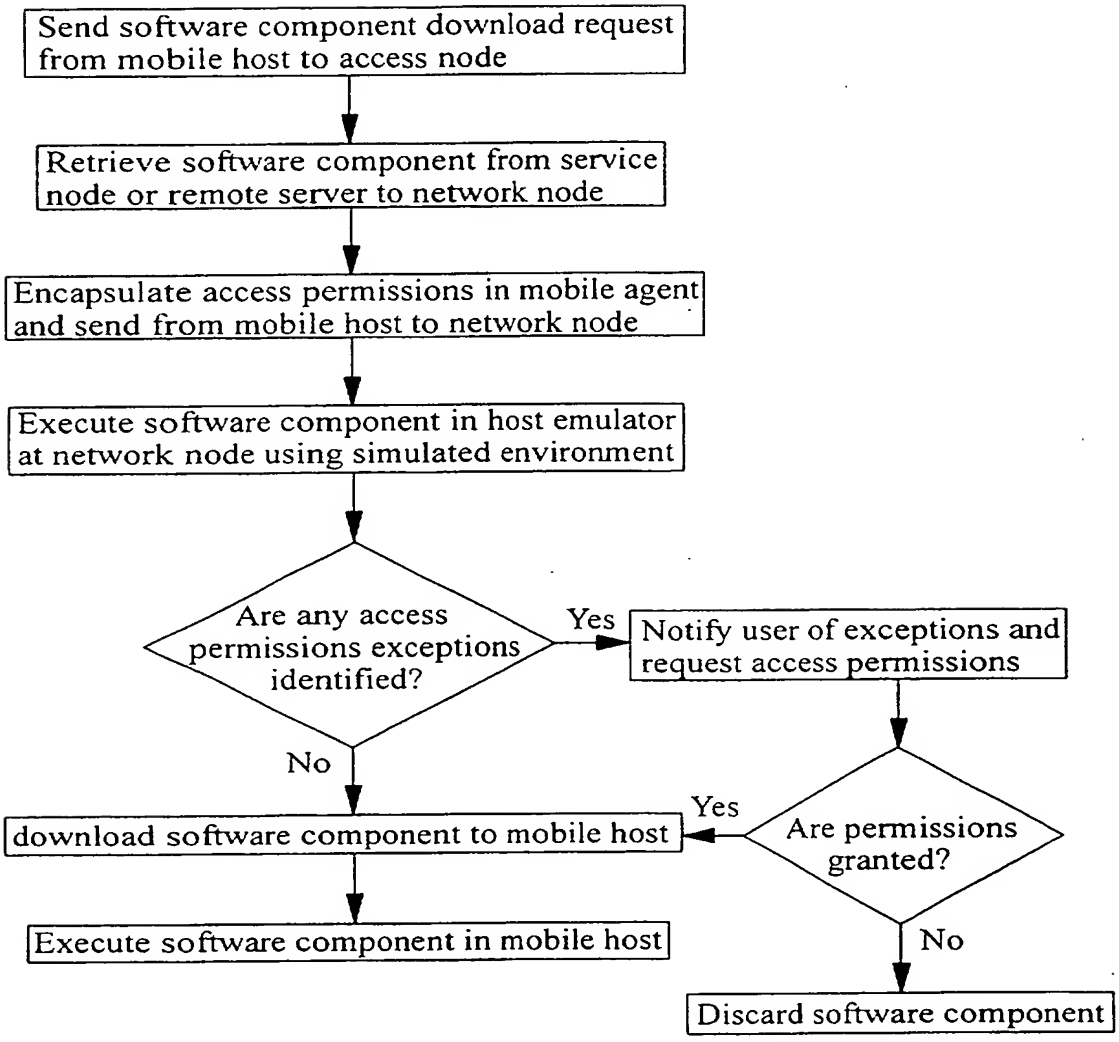
Send software component download request from mobile host to access node
Retrieve software component from service node or remote server to network node
Encapsulate access permissions in mobile agent and send from mobile host to network node
Execute software component in host emulator at network node using simulated environment
Are any access permissions exceptions identified?
Yes
Notify user of exceptions and request access permissions
No
Are permissions granted?
Yes
download software component to mobile host
No
Execute software component in mobile host
Discard software component

Figure 3

# ACCESS RIGHTS IN A MOBILE COMMUNICATIONS SYSTEM

## Field of the Invention

The present invention relates to access rights in a mobile communications system and more particularly to a method and apparatus for allowing the access permissions, required by an executable software component to run in a mobile host, to be determined prior to full execution of the component.

## Background to the invention

Recent years have seen a rapid growth in the use of mobile telecommunications systems. This growth has been accompanied by an ever increasing range of services and functions available to mobile telecommunication network subscribers. At least a proportion of these added services and functions have been made available by the possibility to download software components from a mobile network to a mobile host (e.g. telephone or communicator) over the air interface. Typically, downloaded components are executable in the mobile host.

The volume and range of downloadable software components is likely to grow rapidly in the near future due at least in part to the introduction of new mobile telecommunication standards such as GPRS and UMTS. The introduction of added communication interfaces (such as BLUETOOTH) for facilitating communication between a mobile host and devices such as televisions, vending machines, etc, over a local network is likely to add further to this growth.

Execution of many of the proposed downloadable software components will result in costs being incurred, host functionality being altered, private files being accessed and confidential information disclosed, or other actions being carried out which are likely to be of concern to a subscriber. It is important therefore that a subscriber be able to maintain control over the types of actions which a software component may carry out.

In practice, this is likely to be achieved by attaching to each software component a certificate which is known to the mobile host and which authenticates the source and/or nature of the component to the host. It is also possible that the software component may be accompanied by a set of required "access permissions" which identify to the mobile host the actions which will be taken by the component when it is executed. The required access permissions are likely to be "encapsulated" in the certificate and will be compared by the host against a set of pre-set (or granted) access permissions defined by the user.

Summary of the Present Invention.

The inventor of the present invention has recognised that a mobile host user may wish to execute a downloaded software component even if the component is accompanied by a permission or set of permissions which have not been pre-set at the host . He has further recognised that to execute such a downloaded component in the host may result in faults at the host if it attempts to perform actions which are not permitted or if it requires data which is unavailable, i.e. if the component gives rise to security exceptions.

It is an object of the present invention to overcome or at least mitigate the disadvantage noted in the preceding paragraph. This and other objects are achieved by providing a host emulator in which the executable software component may be run or tested in isolation.

According to a first aspect of the present invention there is provided a method of testing an executable software component which may be downloaded over a mobile telecommunications network to a mobile host for execution in the mobile host, the method comprising:

transferring a copy of the executable software component from a memory in which it resides to an emulator, the emulator being able to at least substantially emulate the operation of a mobile host which wishes to run the software component; and

executing the software component in the emulator and identifying any security exceptions resulting from execution of the software component.

Embodiments of the present invention allow the "testing" of a software component prior to its real time execution in a mobile host. This allows errors and insufficiencies to be predicted and therefore faults, including run-time errors, to be avoided.

Preferably, the method comprises the step of identifying access permissions which are not pre-set for the destination mobile host and which are required to avoid said security exceptions. The mobile host is notified of these identified access permissions, and the host user may be given the opportunity to authorise or deny the identified permissions.

In certain embodiments of the present invention the emulator may be located at the mobile host which wishes to run the software component. In other embodiments, the emulator may be located within the mobile telecommunications network, e.g. at a network node.

Preferably, the host emulator has access to a set of commonly occurring call-stack scenarios which enable the emulator to execute the software component in a suitable simulated environment.

The host emulator is preferably provided with details of the currently defined or pre-set access permissions for the mobile host (or associated subscriber). These may be permanently available to the emulator or may be provided to the emulator each time a request is made to download a software component.

Preferably, said software component is transferred together with a set of required access permissions. The method of the invention may comprise an initial step of comparing the required permissions against the pre-set permissions of the destination host. The emulation step is only carried out if one or more of the required permissions are not pre-set in the host. In certain embodiments of the invention, the required access permissions

may be contained in an authentication certificate attached to or contained in the software component.

Examples of executable software components which may be tested with the present invention include Java written service code and Java applets.

According to a second aspect of the present invention there is provided apparatus for testing an executable software component which may be downloaded over a mobile telecommunications network to a mobile host for execution in the mobile host, the apparatus comprising:

means for transferring a copy of the executable software component from a memory in which it resides to an emulator, the emulator being able to at least substantially emulate the operation of a mobile host which wishes to run the software component; and

means for executing the software component in the emulator and identifying security exceptions resulting from the execution of the software component.

According to a third aspect of the present invention there is provided a mobile host comprising:

a memory arranged in use to store a set of access permissions;

means for executing a software component which may be downloaded into said memory from a remote location over a mobile telecommunications network;

means for making said access permissions available to an emulator which is able to at least substantially emulate the operation of said executing means, the emulation means being arranged in use to receive and execute said software component for the purpose of identifying security exceptions resulting from the execution of the software component; and

means for receiving from the host emulator an identification of access permissions required to overcome any security exceptions identified by the emulator.

In certain embodiments of the present invention, the mobile host comprises means for providing said host emulator. This means may comprise for example a microprocessor which also provides said means for executing the software component.

In other embodiments of the present invention, the host emulator is provided by a means located within the mobile telecommunications network.

According to a fourth aspect of the present invention there is provided a method of testing an executable software component which may be downloaded over a mobile telecommunications network to a mobile host for execution in the mobile host, the method comprising:

analysing the source code of the software component prior to executing it in the mobile host to identify any security exceptions likely to result from execution of the component; and

identifying the access permission(s) required to overcome the security exceptions.

In certain embodiments of the present invention, the step of analysing the source code of the software component is carried out at the mobile host. In other embodiment of the invention, that step is carried out at a node of the mobile telecommunications network.

According to a fifth aspect of the present invention there is provided apparatus for testing an executable software component which may be downloaded over a mobile telecommunications network to a mobile host for execution in the mobile host, the apparatus comprising:

means for analysing the source code of the software component prior to executing it in the mobile host to identify any security exceptions likely to result from execution of the component; and

means for identifying the access permission(s) required to overcome the security exceptions.

Brief Description of the Drawings

Figure 1 illustrates schematically a telecommunications system;
Figure 2 is a flow diagram illustrating a method of testing and downloading an executable software component over the telecommunications network of Figure 1; and
Figure 3 is a flow diagram illustrating an alternative method of testing and downloading an executable software component over the telecommunications network of Figure 1.

Detailed Description of Preferred Embodiments

There is illustrated in Figure 1 a telecommunications system comprising a mobile telecommunications network 1. This network may for example be a Universal Mobile Telecommunications System (UMTS) network which supports both voice and data services using packet switched data transmission. The architecture and functionality of a UMTS network will not be described in detail here but rather reference should be made to the recommendations and standards of the European Telecommunications Standards Institute (ETSI). Figure 1 illustrates an access node 2 of the UMTS network 1 which acts as a local switching centre for mobile hosts such as the host 3 illustrated in the Figure (intermediate base stations and base station controllers may couple the mobile host 3 to the access node 2). Figure 1 also illustrates a gateway node 4 of the UMTS network 1 which represents a gateway between the UMTS network 1 and foreign networks. In this example, the gateway node 4 is coupled to the Internet 5.

The mobile host 3 comprises a microprocessor and digital memory (not shown). In normal use, the microprocessor executes software code permanently stored in the memory to enable the host 3 to perform general purpose operations, e.g. telephone services, Short Message Service, as well as local services such as updating time and calendar information, diary, etc. In addition to these operations, the mobile host 3 is capable of performing additional tasks when appropriate executable software components are downloaded into its memory. These additional operations could include for example an ability to automatically update a host diary with local sporting/arts events over the wireless interface. The configuration of the mobile host 3 may also be altered by downloading executable software components.

The user of the mobile host 3 is able to request from the UMTS network 1 the downloading of specified software components, for example using a display and keypad of the host 3 or using voice commands. In some cases, a menu of software components available for downloading may be sent to the mobile host 3 by the access node 2. The software components may reside either within the UMTS network 1 or outside of that network and may be Java applets (or .jar files). Figure 1 illustrates a service node 6 of the UMTS network 1 and which provides a repository for software components. Software components may alternatively be retrieved, via the gateway node 4, from a remote server 7 which is coupled to the Internet 5. In either case, a requested executable software component is received by the access node 2.

It will be appreciated that the execution of a software component in the mobile host 3 may cause alterations in the state and operability of the mobile host, and may result in additional costs being incurred to the host's user (e.g. as a result of calls being initiated or data being transferred). It is therefore important for the user to maintain control of what "access permissions" are granted to executable software components. For this purpose, the mobile host 3 maintains in its memory a list of access permissions. For example, the user may set an access permission which allows entries to be made into his host's electronic diary, the sending of data messages, etc. If an access permission is not set, then an executed software component requiring that permission to run will most likely fail during execution. Following the downloading of a software component into the memory of the mobile host 3, the required access permissions, which accompany the component (for example contained within an authentication certificate), are inspected by the host to determine if they are all currently set by the hosts user. If the answer is yes, then the component can be executed by the host's microprocessor in the normal manner.

In order to avoid possible run-time errors, components which are accompanied by required access permissions not pre-set by the user are tested in a host emulator following the downloading of the component from the access node 2 into the memory of the mobile host 3. The emulator runs on the mobile host's microprocessor according to

software code pre-stored in the host's memory. The emulator is not allowed to reconfigure any of the host's settings, nor is it allowed to initiate any external communications, e.g. with the UMTS network 1. Rather, the memory contains simulated data (for example experimentally derived approximations regarding access requests) which allows the emulator to run in isolation, i.e. the emulator acts as a "sandbox". When a downloaded software component is executed in the emulator, the emulator identifies all security exceptions raised by non-permitted access requests attempted during execution.

Assuming that one or more such security exceptions are identified, these are interpreted and the access permissions required to overcome the exceptions are presented to the user, for example by displaying them on the host's display. The user then has the opportunity to grant the relevant permissions, in which case the component will be executed for real, or to deny the permissions in which case the component will be discarded. Additional permissions may be granted on a permanent or temporary basis.

The testing and downloading method described above is further illustrated in the flow diagram of Figure 2.

The flow diagram of Figure 3 illustrates an alternative method of testing a software component prior to executing the component in real time in the mobile host 3. In this scenario, the mobile host 3 delegates the emulation task to a network node 8 of the UMTS network 1, the network node 8 being associated with the access node 2 and comprising a suitable microprocessor and memory containing emulation code. The mobile host 3 encapsulates its (use defined) access permissions in a so-called mobile "agent" and sends these to the network node 8. When the host 3 requests the downloading of a software component via the access node 2, the software component is initially routed from the service node 6, or an external node such as the remote Internet server 7, to the network node 8. Assuming that the software component is accompanied by access permissions which are not pre-set by the host user, the emulator is run on the network node 8.

The network node 8 is provided with a large set of commonly occurring call-stack scenarios (of service runs) which have been experimentally derived using test runs. The test runs may have concentrated on typical service types offered in the majority of networks. The network node 8 runs the host emulator using as its operating environment the access permissions received in the mobile agent and the call-stack scenarios. As with the previous embodiment, any security exceptions identified during the emulation are identified and interpreted to determine the required access permissions. These permissions are then presented to the user (by encapsulating them in a mobile agent which is returned to the host 3), who then has the opportunity to grant or deny the relevant permissions.

The advantages of this second embodiment include the greater memory and processing power available at a network node which allow for faster emulation with a greater number of predefined call-stack scenarios, as well as a reduction in the computational load placed on the mobile host. A disadvantage is a potential increase in the volume of signalling traffic.

It will be appreciated by the person of skill in the art that various modifications may be made to the above described embodiments without departing from the scope of the present invention. For example, security exceptions may be identified by analysing the associated code (if available) prior to its execution in the mobile host 3 to identify the required permissions.

CLAIMS:

1. A method of testing an executable software component which may be downloaded over a mobile telecommunications network to a mobile host for execution in the mobile host, the method comprising:

transferring a copy of the executable software component from a memory in which it resides to an emulator, the emulator being able to at least substantially emulate the operation of a mobile host which wishes to run the software component; and

executing the software component in the emulator and identifying any security exceptions resulting from execution of the software component.

2. A method according to claim 1 and comprising the step of identifying access permissions which are not pre-set for the destination mobile host and which are required to avoid said security exceptions.

3. A method according to claim 2 and comprising notifying the mobile host of these identified access permissions, and giving the host user the opportunity to authorise or deny the identified permissions.

4. A method according to any one of the preceding claims, wherein the emulator is located at the mobile host which wishes to run the software component.

5. A method according to any one of claims 1 to 3, wherein the emulator is located within the mobile telecommunications network.

6. A method according to any one of the preceding claims, wherein the host emulator has access to a set of commonly occurring call-stack scenarios which enable the emulator to execute the software component in a suitable simulated environment.

7. A method according to any one of the preceding claims, wherein the host emulator is provided with details of the currently defined or pre-set access permissions for the mobile host or associated subscriber.

8. A method according to any one of the preceding claims, wherein said software component is transferred together with a set of required access permissions.

9. A method according to claim 8 and comprising an initial step of comparing the required permissions against the pre-set permissions of the destination host and carrying out the emulation step only if one or more of the required permissions are not pre-set in the host.

10. Apparatus for testing an executable software component which may be downloaded over a mobile telecommunications network to a mobile host for execution in the mobile host, the apparatus comprising:

means for transferring a copy of the executable software component from a memory in which it resides to an emulator, the emulator being able to at least substantially emulate the operation of a mobile host which wishes to run the software component; and

means for executing the software component in the emulator and identifying security exceptions resulting from the execution of the software component.

11. A mobile host comprising:

a memory arranged in use to store a set of access permissions;

means for executing a software component which may be downloaded into said memory from a remote location over a mobile telecommunications network;

means for making said access permissions available to an emulator which is able to at least substantially emulate the operation of said executing means, the emulation means being arranged in use to receive and execute said software component for the purpose of identifying security exceptions resulting from the execution of the software component; and

means for receiving from the host emulator an identification of access permissions required to overcome any security exceptions identified by the emulator.

12. A mobile host according to claim 11, wherein the mobile host comprises means for providing said host emulator.

13. A mobile host according to claim 11, wherein the host emulator is provided by a means located within the mobile telecommunications network.

14. A method of testing an executable software component which may be downloaded over a mobile telecommunications network to a mobile host for execution in the mobile host, the method comprising:

analysing the source code of the software component prior to executing it in the mobile host to identify any security exceptions likely to result from execution of the component; and

identifying the access permission(s) required to overcome the security exceptions.

15. A method according to claim 14, wherein the step of analysing the source code of the software component is carried out at the mobile host.

16. A method according to claim 14, wherein the step of analysing the source code of the software component is carried out at a node of the mobile telecommunications network.

17. Apparatus for testing an executable software component which may be downloaded over a mobile telecommunications network to a mobile host for execution in the mobile host, the apparatus comprising:

means for analysing the source code of the software component prior to executing it in the mobile host to identify any security exceptions likely to result from execution of the component; and

means for identifying the access permission(s) required to overcome the security exceptions.

The Patent Office

INVESTOR IN PEOPLE

**Application No:** GB 9920703.7 **Examiner:** John Betts
**Claims searched:** 1-13 **Date of search:** 8 March 2000

# Patents Act 1977
# Search Report under Section 17

**Databases searched:**

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK Cl (Ed.R): H4L (LDGX, LED) G4A (AFL)

Int Cl (Ed.7): H04Q 7/32, 7/22 G06F 9/445

Other: On-line: WPI, EPODOC, JAPIO

**Documents considered to be relevant:**

| Category | Identity of document and relevant passage | Relevant to claims |
|---|---|---|
| A | EP0930793 A (Texas Instr) see description relevant to Figs 5-6 | |
| A | EP0767426 A (Siemens) see abstract | |
| A | WO96/24231 A (Ericsson) see description of fig3(a) | |

| | | | |
|---|---|---|---|
| X | Document indicating lack of novelty or inventive step | A | Document indicating technological background and/or state of the art. |
| Y | Document indicating lack of inventive step if combined with one or more other documents of same category. | P | Document published on or after the declared priority date but before the filing date of this invention. |
| & | Member of the same patent family | E | Patent document published on or after, but with priority date earlier than, the filing date of this application. |